
LIBRARY

```
  OOOOOO    TTTTTTTTTT   SSSSSSSS   CCCCCCCC  VV      VV  TTTTTTTTTT  TTTTTTTTTT  LL          LL
  OOOOOO    TTTTTTTTTT   SSSSSSSS   CCCCCCCC  VV      VV  TTTTTTTTTT  TTTTTTTTTT  LL          LL
OO      OO      TT      SS         CC        VV      VV      TT          TT      LL          LL
OO      OO      TT      SS         CC        VV      VV      TT          TT      LL          LL
OO      OO      TT      SS         CC        VV      VV      TT          TT      LL          LL
OO      OO      TT      SS         CC        VV      VV      TT          TT      LL          LL
OO      OO      TT        SSSSSS   CC        VV      VV      TT          TT      LL          LL
OO      OO      TT        SSSSSS   CC        VV      VV      TT          TT      LL          LL
OO      OO      TT              SS CC        VV      VV      TT          TT      LL          LL
OO      OO      TT              SS CC         VV    VV       TT          TT      LL          LL
OO      OO      TT              SS CC         VV    VV       TT          TT      LL          LL
  OOOOOO        TT      SSSSSSSS   CCCCCCCC     VV          TT          TT      LLLLLLLLLL  LLLLLLLLLL
  OOOOOO        TT      SSSSSSSS   CCCCCCCC     VV          TT          TT      LLLLLLLLLL  LLLLLLLLLL

LL            IIIIII     SSSSSSSS
LL            IIIIII     SSSSSSSS
LL              II     SS
LL              II     SS
LL              II     SS
LL              II     SS
LL              II       SSSSSS
LL              II       SSSSSS
LL              II             SS
LL              II             SS
LL              II             SS
LL              II             SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

Table of contents

(2) 50 DECLARATIONS
(3) 82 OTS$CVT_TL_L - Convert text (logical) to longword

```
0000     1          .TITLE  OTS$CVTTLL - Convert text (logical) to longword
0000     2          .IDENT  /1-003/                         ; File: OTSCVTTLL.MAR  Edit: SBL1003
0000     3
0000     4  ;
0000     5  ;****************************************************************************************
0000     6  ;*                                                                                      *
0000     7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                             *
0000     8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                              *
0000     9  ;*  ALL RIGHTS RESERVED.                                                                *
0000    10  ;*                                                                                      *
0000    11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED               *
0000    12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE              *
0000    13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER               *
0000    14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY               *
0000    15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY               *
0000    16  ;*  TRANSFERRED.                                                                        *
0000    17  ;*                                                                                      *
0000    18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE               *
0000    19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT              *
0000    20  ;*  CORPORATION.                                                                        *
0000    21  ;*                                                                                      *
0000    22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS              *
0000    23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                             *
0000    24  ;*                                                                                      *
0000    25  ;*                                                                                      *
0000    26  ;****************************************************************************************
0000    27  ;
0000    28
0000    29  ;++
0000    30  ; FACILITY: Language independent support library
0000    31  ;
0000    32  ; ABSTRACT:
0000    33  ;
0000    34  ; A routine to convert a text string indicating "true" or "false" to
0000    35  ; a integer representation.
0000    36  ;
0000    37  ; ENVIRONMENT: User Mode, AST Reentrant
0000    38  ;
0000    39  ;--
0000    40  ; AUTHOR: Steven B. Lionel, CREATION DATE: 01-Mar-1979
0000    41  ;
0000    42  ; MODIFIED BY:
0000    43  ;
0000    44  ; Edit History
0000    45
0000    46  ; 1-001 - Original.  Complete rewrite of FOR$CNV_IN_L.  SBL 01-Mar-1979
0000    47  ; 1-002 - SBL24781  Fix bug where SP is used instead of AP.  SBL 2-Jul-79
0000    48  ; 1-003 - Do correct thing if value_size is wrong.  SBL 25-Feb-1980
```

```
                        50          .SBTTL  DECLARATIONS
                        51 ;
                        52 ; INCLUDE FILES:
                        53 ;
                        54
                        55 ;
                        56 ; EXTERNAL DECLARATIONS:
                        57 ;
                        58          .DSABL  GBL                             ; Prevent undeclared
                        59                                                  ; symbols from being
                        60                                                  ; automatically global.
                        61          .EXTRN  OTS$_INPCONERR                  ; Error code
                        62
                        63 ;
                        64 ; MACROS:
                        65 ;
                        66
                        67 ;
                        68 ; EQUATED SYMBOLS:
                        69 ;
                        70
                        71 ;
                        72 ; OWN STORAGE:
                        73 ;
                        74
                        75 ;
                        76 ; PSECT DECLARATIONS:
                        77 ;
                        78          .PSECT _OTS$CODE PIC, USR, CON, REL, LCL, SHR, -
                        79                      EXE, RD, NOWRT, LONG
                        80
```

```
                        .SBTTL  OTS$CVT_TL_L - Convert text (logical) to longword
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;       This routine converts a text string to a longword value
                ;       using FORTRAN-77 type L format conversion.  The format of
                ;       the text accepted is as follows:
                ;
                ;               < 0 or more blanks >
                ;               <       < end of string >
                ;                       or
                ;                       <       < "." or nothing >
                ;                       Letter: < "T", "t", "F", "f" >
                ;                               < 0 or more of any character >
                ;                               < end of string > > >
                ;       The value returned by OTS$CVT_TL_L is:
                ;
                ;               All 1 bits (-1) if the character
                ;                   denoted "Letter" above is "T" or "t".
                ;               Zero otherwise.
                ;
                ;       If the input string does not conform to the above specification,
                ;       value is set to zero (false), and the condition code
                ;       OTS$_INPCONERR is returned.
                ;
                ;       Note:  ANSI X3.9-1978 FORTRAN-77 does not allow for the input
                ;       field to be all blank, but since VAX-11 FORTRAN IV-PLUS did
                ;       allow it, (returning .FALSE.), it is allowed here too.
                ;
                ;       Note:  With this implementation, the strings ".TRUE." and
                ;       ".FALSE." are valid.  In the previous version, they were not.
                ;
                ;       Note: For compatibility with previous releases, the global
                ;       symbol FOR$CNV_IN_L may be used to call OTS$CVT_TL_L.
                ;
                ;
                ; CALLING SEQUENCE:
                ;
                ;       status.wlc.v = OTS$CVT_TL_L (input_string.rt.ds, value.wl.r
                ;                                    [, value_size.rl.v])
                ;
                ; INPUT PARAMETERS:
                ;
00000004 0000   ;       input_string    = 4     ; Input text string by descriptor
0000000C 0000   ;       value_size      = 12    ; Size of value in bytes.  If not
                ;                               ; specified, 4 is assumed.  Valid
                ;                               ; values are 1, 2 and 4.  If invalid,
                ;                               ; an error is returned.
                ;                               ; If size is greater than 4, 4 is assumed.
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       NONE
                ;
                ; OUTPUT PARAMETERS:
                ;
00000008 0000   ;       value           = 8     ; Output value by reference
```

```
                                0000   139 ;
                                0000   140 ; IMPLICIT OUTPUTS:
                                0000   141 ;
                                0000   142 ;       NONE
                                0000   143 ;
                                0000   144 ; COMPLETION CODES:
                                0000   145 ;
                                0000   146 ;       SS$_NORMAL      - Successful completion
                                0000   147 ;       OTS$_INPCONERR  - Input conversion error
                                0000   148 ;
                                0000   149 ; SIDE EFFECTS:
                                0000   150 ;
                                0000   151 ;       NONE
                                0000   152 ;
                                0000   153 ;--
                                0000   154
                                0000   155 FOR$CNV_IN_L::               ; For compatibility
                          0004  0000   156         .ENTRY  OTS$CVT_TL_L, ^M<R2>
                                0002   157
                          52    D4  0002   158         CLRL    R2                      ; Initial value is .FALSE.
                     50   04 BC    7D  0004   159         MOVQ    @input_string(AP), R0   ; Get descriptor
                61   50   20    3B  0008   160         SKPC    #^A/ /, R0, (R1)        ; Skip blanks
                          34    13  000C   161         BEQL    FALSE                   ; All blank, .FALSE.
                     61   2E    91  000E   162         CMPB    #^A/./, (R1)            ; A period?
                          06    12  0011   163         BNEQ    10$                     ; No
                          50    D7  0013   164         DECL    R0                      ; Yes, decrement character count
                          1A    13  0015   165         BEQL    ERROR                   ; Last character, error
                          51    D6  0017   166         INCL    R1                      ; Skip over period
                     61   54 8F    91  0019   167 10$:    CMPB    #^A/T/, (R1)            ; Test for .TRUE.
                          1B    13  001D   168         BEQL    TRUE
                     61   74 8F    91  001F   169         CMPB    #^A/t/, (R1)
                          15    13  0023   170         BEQL    TRUE
                     61   46 8F    91  0025   171         CMPB    #^A/F/, (R1)            ; Test for .FALSE.
                          17    13  0029   172         BEQL    FALSE
                     61   66 8F    91  002B   173         CMPB    #^A/f/, (R1)
                          11    13  002F   174         BEQL    FALSE
                                0031   175
                                0031   176 ;+
                                0031   177 ; Invalid character, return error OTS$_INPCONERR
                                0031   178 ;-
50   00000000'                8F    D0  0031   179 ERROR:  MOVL    #OTS$_INPCONERR, R0     ; Error status code
                          0D    11  0038   180         BRB     EXIT                    ; Return with value .FALSE.
                                003A   181
                                003A   182 ;+
                                003A   183 ; Text string represents .TRUE.
                                003A   184 ;-
                          52    01  CE  003A   185 TRUE:   MNEGL   #1, R2                  ; Value is .TRUE.
                          50    01  D0  003D   186         MOVL    #1, R0                  ; SS$_NORMAL
                                05    11  0040   187         BRB     EXIT                    ; Exit
                                0042   188
                                0042   189 ;+
                                0042   190 ; Text string represents .FALSE.
                                0042   191 ; R2 is already cleared.
                                0042   192 ;-
                          50    01  D0  0042   193 FALSE:  MOVL    #1, R0                  ; SS$_NORMAL
                                00    11  0045   194         BRB     EXIT                    ; Exit
                                0047   195
```

```
                     0047   196 ;+
                     0047   197 ; Return to caller with status value.
                     0047   198 ;-
   03   6C   91      0047   199 EXIT:   CMPB    (AP), #<value_size/4>   ; Is size present?
        1E   19      004A   200         BLSS    40$                     ; No, longword
04   0C   AC   D1    004C   201         CMPL    value_size(AP), #4      ; Is it a longword?
        18   18      0050   202         BGEQ    40$                     ; Yes, at least
02   0C   AC   D1    0052   203         CMPL    value_size(AP), #2      ; Is it a word?
        0C   13      0056   204         BEQL    20$                     ; Yes
01   0C   AC   D1    0058   205         CMPL    value_size(AP), #1      ; Is it a byte?
        D3   12      005C   206         BNEQ    ERROR                   ; If not, it's an error
08 BC   52   90      005E   207         MOVB    R2, @value(AP)          ; Convert byte
        0A   11      0062   208         BRB     50$                     ; Exit
08 BC   52   B0      0064   209 20$:    MOVW    R2, @value(AP)          ; Move a word
        04   11      0068   210         BRB     50$                     ; Exit
08 BC   52   D0      006A   211 40$:    MOVL    R2, @value(AP)          ; Move a longword
             04      006E   212 50$:    RET                             ; Return to caller
                     006F   213
                     006F   214         .END
```

```
ERROR             00000031 R     01
EXIT              00000047 R     01
FALSE             00000042 R     01
FOR$CNV_IN_L      00000000 RG    01
INPUT_STRING    = 00000004
OTS$CVT_TL_L      00000000 RG    01
OTS$_INPCONERR    ******** X     00
TRUE              0000003A R     01
VALUE           = 00000008
VALUE_SIZE      = 0000000C
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
_OTS$CODE	0000006F (111.)	01 (1.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.05	00:00:02.10
Command processing	115	00:00:00.31	00:00:03.38
Pass 1	71	00:00:00.31	00:00:02.08
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	52	00:00:00.29	00:00:00.89
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	2	00:00:00.01	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	274	00:00:00.98	00:00:08.47

```
The working set limit was 750 pages.
2694 bytes (6 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 10 non-local and 4 local symbols.
214 source lines were read in Pass 1, producing 11 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.
```

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

```
0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:OTSCVTTLL/OBJ=OBJ$:OTSCVTTLL MSRC$:OTSCVTTLL/UPDATE=(ENH$:OTSCVTTLL)
```

OTSCVTPD LIS

OTSCVTTF LIS

OTSCVTRFP LIS

OTSCVTRT LIS

OTSCVTTOL LIS

OTSCVTPG LIS

OTSCVTTR LIS

OTSCVTRHP LIS

OTSLINKAG LIS

OTSCVTRDP LIS

OTSCVTTIL LIS

OTSCVTTLL LIS

OTSCVTPF LIS

OTSCVTRGP LIS

OTSMOVE LIS

OTSMSG LIS

OTSCVTPH LIS

OTSLUN LIS